

- (c) Extended=>Between
- (d) Between=>Retracted
- (3) Outputs
 - (a) Extend (both valves opened=4 outputs high)
 - (b) Retract (main valve closed=2 outputs high)
- (4) Inputs
 - (a) Retracted (retract proximity sensors on for all cylinders)
 - (b) Extended (extend proximity sensors off for all cylinders)
 - (c) Between (one or more sets of proximity sensors both off)
 - (d) Fault 1 (one set of proximity sensors both on)
 - (e) Fault 2 (one or more of the set of sensors disagrees with the others for a nominally significant time period).
- 13) Diagnostics
 - a) Definition: Status-based diagnostics—specifies the step (s) that the machine is currently waiting to occur (if a fault occurs it specifies the step(s) that were waiting to occur at the time of the fault, i.e., the symptoms).
 - Note: this information is available for both well behavior and fault behavior.
 - b) Definition2: Causal model-based diagnostics—use a model of causal relationships to develop rules that relate machine status to root causes.
 - c) Examples:
 - i) Consider that a human mechanic has incorrectly moved the mount location of a part present proximity sensor causing a misalignment.
 - (1) Status-based: determines that the machine is “waiting for part present sensor #2” (no automatic inference possible)
 - ii) Consider that the proximity sensor on a clamp cylinder has failed
 - (1) Status-based: determines that machine is “waiting for clamp cylinder 2504A”
 - (2) Causal model-based: logic infers that the extend proximity sensor on cylinder 2504A has failed, or that cylinder 2504A is stuck.
- 14) Schematics
 - a) Definition: a schematic is a representation of the electrical, pneumatic, and hydraulic interface to the control assembly.
 - b) Examples:
 - i) Ielectrical
 - ii) Ipneumatic
 - iii) Ihydraulic
 - iv) . . .
- 15) Visualization
 - a) Definition: entities within the control assembly that are useful to portray textually or graphically.
 - b) Examples:
 - i) Control Components (textually or graphically)
 - ii) Logic (graphically: LL, Function Blocks, Axis-like diagrams, state diagrams . . .) what ever conveys the logic to the user.
 - iii) Diagnostics
 - (1) Status messages
 - (2) Causal messages
 - iv) Schematics
 - (1) Typed connections (electrical, pneumatic, network, . . .) within Control Assembly and to and from the Control Assembly (i.e., the interface to the Control Assembly).
 - (2) Bill of Materials (to deploy the control assembly)
 - (3) . . .

- v) Controlled Resources
- vi) Requests
- 16) Controlled Resources
 - a) Definition: properties of the resource controlled by the control assembly that place requirements (i.e., add constraints) on the structure of the assembly
 - b) Example
 - i) Clamp 1
 - (1) Safety constraint: if lose power clamp must fail open
- 17) Requests or Conditions
 - a) Definition: request an operation (optionally with confirmation) or request a status of the external world (color code)
 - i) Request Action Request Status
 - ii) Request Action
 - iii) Request Status
 - iv) Note: how to handle complicated actions (initialization, robot protocols, . . .)
 - b) Examples:
 - i) PartPresent
 - (1) SensePart (Request Status)
 - ii) ClearToEnterLight
 - (1) TurnOn (Request Action)
 - (2) TurnOff (Request Action)
 - iii) SafeBulkHeadClamp
 - (1) Extend
 - (2) Retract
 - iv) SafetyGate
 - (1) SenseSafe (Request Status)
- 18) Documentation

While the invention is described in terms of preferred embodiments in a specific system environment, those skilled in the art will recognize that the invention can be practiced, with modification, in other and different hardware and software environments within the spirit and scope of the appended claims.

Having thus described our invention, what we claim as new, and desire to secure by Letters Patent is:

 1. A method of enabling a computer having a display and operating under control of an object oriented framework, including one or more control assemblies, to control an enterprise, comprising the steps of:
 - a) defining an interface for retrieving a model from an external source;
 - b) storing information in the external source which complies with the interface;
 - c) retrieving the model from the external source; and
 - d) generating an enterprise control system utilizing the model wherein the external source is transformed from a flow chart to a timing diagram.
 2. A system, including a computer having a display and operating under the control of an object oriented framework, including a control assembly, to control an enterprise, comprising:
 - a) a first code segment that defines an interface for retrieving a model from an external source;
 - b) a second code segment that stores information in the external source which complies with the interface;
 - c) a third code segment that retrieves the model from the external source; and
 - d) a fourth code segment that generates an enterprise control system utilizing the model;
 wherein the external source is transformed from a flow chart to a timing diagram.